measurement SPECIALTIES

PRESSURE SYSTEMS

Model 9046
Intelligent Temperature Scanner
User's Manual

August 2008

NetScanner™ System

www.PressureSystems.com

©This User's Manual is a copyright product of Pressure Systems, Inc., 2008

Permission is hereby granted to make copies and distribute verbatim copies of
this manual, provided the copyright notice and this permission notice are preserved on all copies.

Table of Contents

1.0 Introduction 1
1.1 General 1
1.1.1 Introduction 1
1.1.2 Physical Description 2
1.1.3 Theory of Operation 3
1.1.3.1 Thermocouple Cold Junction Compensation 3
1.1.3.2 Open Circuit Detection 4

2.0 Setup Instructions 4
2.1 Equipment Required 5
2.2 Computer Preparation 6
2.3 Module Connection 8
2.4 IP Address Assignment 10
2.5 Set Sensor Type 14
2.6 Read Sensor Type 16
2.7 Thermocouple Reference Temperature Alarms 16
2.8 Set Temperature Units 18
2.9 Read Temperature Units 18
2.10 Write EU Filter Parameter 18
2.11 Read EU Filter Parameter 20

3.0 Calibration 21

4.0 Firmware 23
4.1 Description of Operation 23
4.2 Updating the Firmware 25

5.0 Command Set 27
5.1 Summary Information About All Commands 29
5.2 No Operation (Command 'A') 32
5.3 Reset (Command 'B') 33
5.4 Read Primary Transducer Volts (Command 'V') 34
5.5 Read Primary Transducer A/D Counts (Command 'a') 36
5.6 Read Primary Transducer E.U. Data at High Speed (Command 'b') 38
5.7 Define/Control Autonomous Host Streams (Command 'c') 40
5.7.1 Sub-command Index 00: Configure a Host Delivery System 43
5.7.2 Sub-command Index 01: Start Stream(s) 48
5.7.3 Sub-command Index 02: Stop Stream(s) 50
5.7.4 Sub-command Index 03: Clear Stream(s) 52
5.7.5 Sub-command Index 04: Return Stream Information 54
5.7.6 Sub-command Index 05: Select Prefix/Data Groups in a Stream 56
5.7.7 Sub-command Index 06: Select Protocol for Stream Delivery 59

Appendix A 63

Our Company

Pressure Systems Incorporated, (PSI) develops, manufactures, and services level and pressure measuring instruments where the highest level of traceable accuracy is required for aerospace, industrial, municipal, and environmental applications. Our products have become the world standard for electronic level and pressure measurement and scanning. We are committed to the highest quality design, manufacture, and support of level and pressure instrumentation that is in the best interest of our customers. PSI is an ISO-9001:2000 certified company.

Our Warranty

Pressure Systems, Inc., warrants **NetScanner™ System** products to be free of defects in material and workmanship under normal use and service for one (1) year.

Technical Support

Monday through Friday, during normal working hours, (7:30 am through 5:30 pm, Eastern time) knowledgeable personnel are available for assistance and troubleshooting. Contact the **Applications Support Group** or the **Customer Services Department** at Pressure Systems (**757-865-1243** or toll free **1-800-328-3665**) if your scanner is not operating properly or if you have questions concerning any of our products. E-mail assistance is available by contacting Applications@PressureSystems.com.

Merchandise Return Procedures

If your scanner needs to be returned to Pressure Systems, obtain a Returned Merchandise Authorization (RMA) from the Customer Service Department.

Be prepared to supply the following information when requesting the RMA:

- Part number
- Serial number
- Complete description of problems/symptoms
- **Bill To** and **Ship To** address
- Purchase order number (not required by PSI warranty repairs)
- Customer contact and telephone number

The above information, including the RMA number must be on the customer's shipping documents that accompany the equipment to be repaired. PSI also requests that the outside of the shipping container be labeled with the RMA number to assist in tracking the repairs. All equipment should be sent to the following address:

ATTN: PSI REPAIR DEPARTMENT (7-digit RMA number)
Pressure Systems, Inc.
34 Research Drive
Hampton, Virginia 23666

PSI will return warranty items prepaid via UPS GROUND. If the customer desires another method of return shipment, PSI will prepay and add the shipping charges to the repair bill.

Incoming freight charges are the customer's responsibility. The customer is also responsible for paying shipping charges to and from PSI for any equipment not under warranty.

All products covered under the PSI warranty policy will be repaired at no charge. An analysis fee will be charged to quote the cost of repairing any item not under warranty. If, for any reason, the customer decides not to have the item repaired, the analysis fee will still be charged. If the quote is approved by the customer, the analysis fee will be waived. The quote for repair will be based on the PSI flat rate for repair, calibration, and board replacement. When these prices do not apply, the quote will be based on an hourly labor rate plus parts. All replaced parts are warranted for 90 days from the date of shipment. The 90-day warranty is strictly limited to parts replaced during the repair.

Website and E-Mail

Visit our website at www.PressureSystems.com to look at our new product releases, application notes, product certifications, and specifications. E-mail your questions and comments to us: Sales@PressureSystems.com.

Our Firmware

This manual was prepared for various versions of module firmware as were released at the time of this manual publication. Addenda will be distributed as deemed necessary by PSI. Any questions regarding firmware upgrades may be addressed to the **Applications Support Group**. Firmware revisions, manual addenda, and utility software may also be obtained from the PSI web page at www.PressureSystems.com.

Our Publication Disclaimer

This document is thoroughly edited and is believed to be thoroughly reliable. Pressure Systems, Inc., assumes no liability for inaccuracies. All computer programs supplied with your products are written and tested on available systems at the factory. PSI assumes no responsibility for other computers, languages, or operating systems. PSI reserves the right to change the specifications without notice.

This page left intentionally blank.

1.0 Introduction

1.1 General

1.1.1 Introduction

The Model 9046 Intelligent Temperature Scanner is a 16 channel modular scanning system designed to operate in harsh environments, such as turbine engine test cells. It uses an Ethernet TCP/IP or UDP interface and has the same mechanical footprint, power and interface connector, and front panel LED indicators as the Model 9116 Intelligent Pressure Scanner. The instruction set is compatible with other NetScanner™ System products.

The module is designed to remotely report temperatures from a variety of sensors. These include thermocouples, Resistance Temperature Detectors, and thermistors. It is also capable of returning resistance values and DC voltages.

Once configured, the Model 9046 will return Engineering Unit temperature readings in degrees Fahrenheit, or degrees Centigrade. It incorporates an adjustable (by channel) running average filter for use with calibrators that have electrically noisy outputs or other signal sources with noisy outputs.

Two input styles are available; direct wired side entry units or top entry miniature thermocouple connector style. Both styles are splash-resistant upon proper installation. The side entry version is CE rated. Power and Ethernet connections are made via one circular connector.

1.1.2 Physical Description

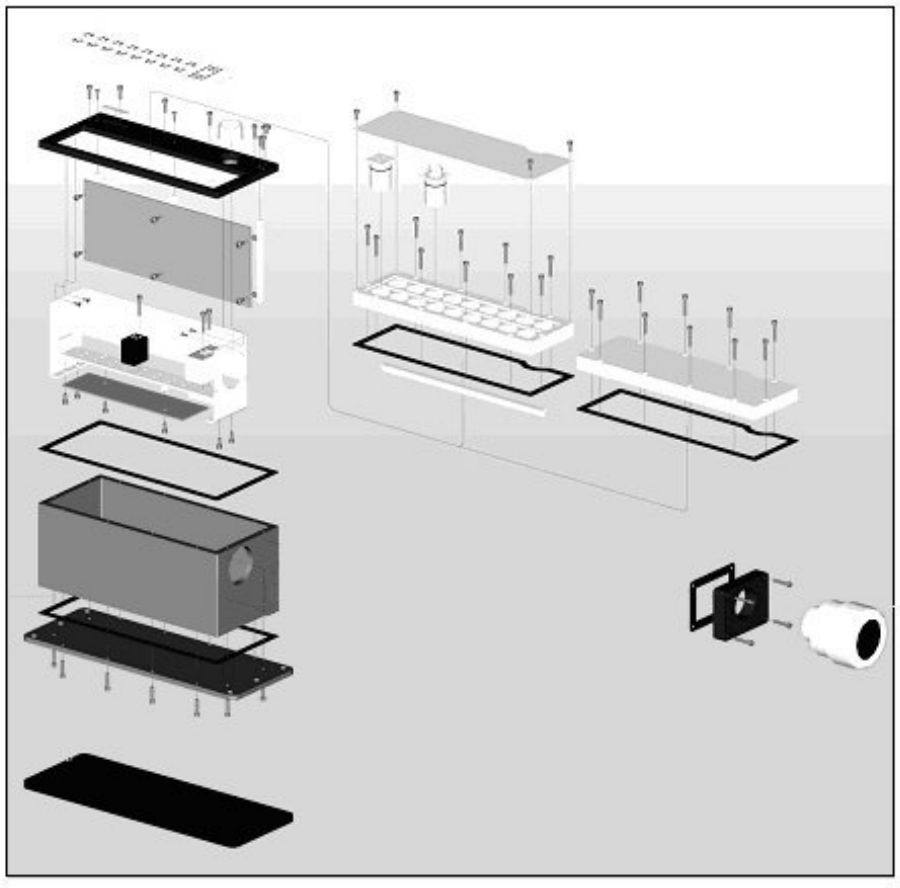


Figure 1-1: Exploded View of Module

The Model 9046 temperature scanner is housed in a rugged machined aluminum case. The case is splash proof and finished to allow installation in a turbine engine test cell environment. Sensors can be connected to the Model 9046 in two different wiring configurations:

- Top panel thermocouple mini-jacks, or MIL style circular connectors (RTDs, thermistors, etc.)
- Side entry for direct connection. (Crouse Hinds CGB box style strain relief on end panel).

The direct wire connection is the most economical, and the most accurate, as it eliminates one thermocouple junction and its attendant potential for errors.

The Model 9046 has the same mounting footprint as the PSI NetScanner family of pressure scanners. It is approved for operation in an ambient temperature environment from –30 to +70 degrees Centigrade.

For operation, the Model 9046 only requires 18 to 36 VDC (i=200mA), and the appropriate Ethernet interface.

1.1.3 Theory of Operation

The Model 9046 is a 32 channel analog multiplexer, controlled by a microprocessor. It communicates over an Ethernet TCP/IP or UDP interface. The module utilizes 16 small Signal Conditioning Modules (SCMs), which plug into a motherboard inside the module. There are two types:

- universal (thermocouple or resistance, set by jumper)
- thermocouple ***without*** open circuit detection

1.1.3.1 Thermocouple Cold Junction Compensation

To accomplish cold junction compensation, the thermocouple SCMs incorporate a precision temperature device located between the thermocouple input sockets. It is in close proximity (1/16" or 1.5 mm) to the actual contacts. This gives very precise junction temperature measurement, which contributes to the overall accuracy of the module.

Sixteen of the multiplexer channels are used to scan the embedded precision temperature devices. The processor then calculates the junction temperature. The temperature this precision temperature device reports (a voltage that will be referred to as Vb) is limit checked, and stored for use in the Engineering Unit (EU) calculation to compensate for junction temperature. Junction temperatures which deviate by a preset amount from the average of the other junction temperatures will issue an alarm to the external host. An alarm will also be issued if the absolute temperature of any junction exceeds the rated operating temperature of the 9046.

The thermocouple channels are then scanned, and the voltages used in the EU calculation. The EU calculation uses previously defined coefficients to determine thermocouple type, engineering unit, and calibration offset. The universal SCM incorporates the thermocouple and resistance circuitry in the same package. A jumper can be installed to tell the SCM to operate in the thermocouple mode. When the jumper is not installed, it will function in the resistance mode. In this mode, the Vb channel is internally connected to measure the voltage drop (Vb) across an internal precision resistor. The processor then solves for line current which is a result of an internal reference voltage being passed through the unknown resistance (RTD Thermistor, Resistor, etc.) Now that the current is known, and the Voltage (Va) is known, the resistance is calculated. By performing a table-lookup, or polynomial calculation, the proper engineering unit value is obtained and returned in a packet via the Ethernet connection.

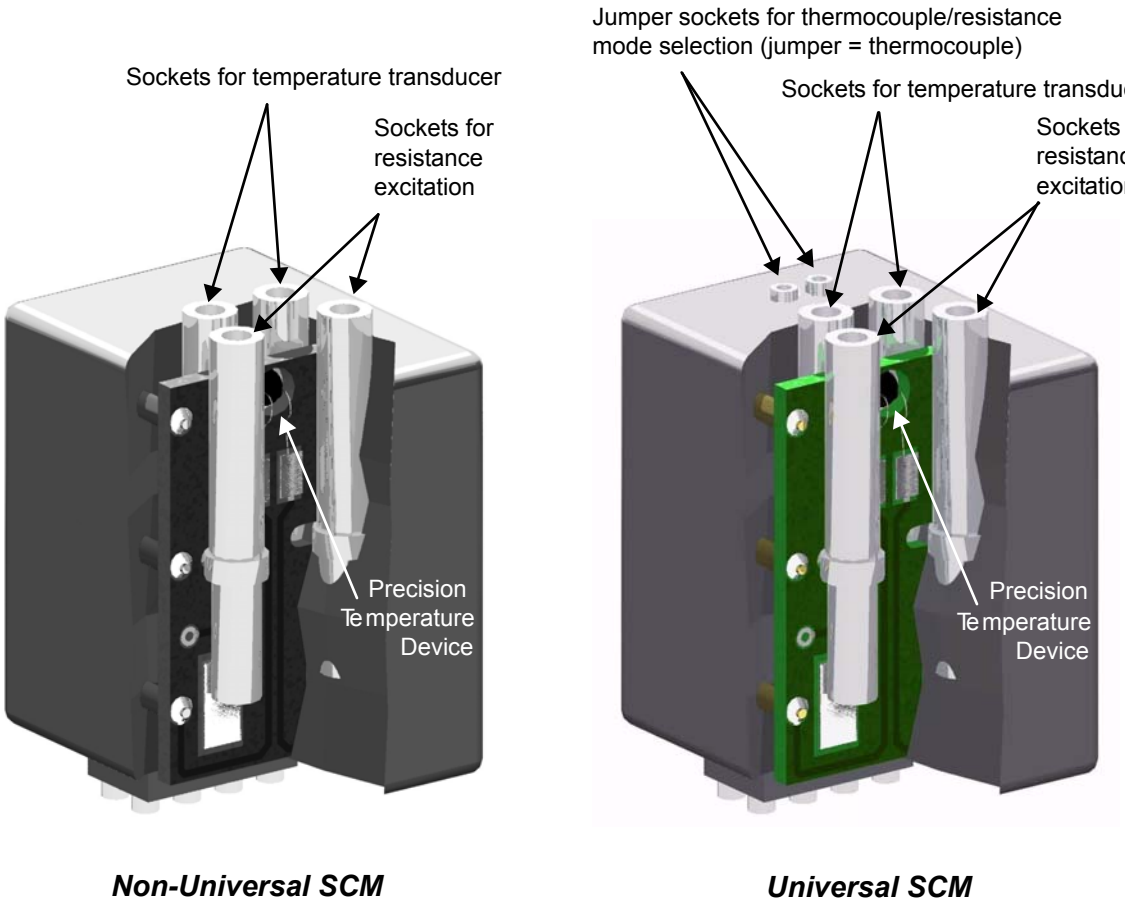


Figure 1-2: Signal Conditioning Module (SCM)

1.1.3.2 Open Circuit Detection

The SCM, without the open circuit detect feature, is used where an electronic engine control package is used which requires a high degree of isolation from ground in order to reduce offsets in the temperature signal. This unit will handle thermocouples only and has no jumper.

2.0 Setup Instructions

Use the procedures in this section to set up a Model 9046 module.

2.1 Equipment Required

The following items will be needed in order to set-up a Temperature Scanning Module:

- Model 9046 Temperature Scanning Module
- PSI powered network solution such as 90DB, 90DC, 98RK or 9FC. 9IFC will be used as the example.
- PSI 9080 cable for the connection between the 9IFC and the Model 9046 module. See Cable Connections.
- Ethernet compatible AAUI transceiver and Ethernet cables
- Personal computer (486 MHz or higher) equipped with:
    - An Ethernet interface
    - Windows® 98, NT, 2000, or XP operating system
    - PSI NUSS software

Note

NUSS software is included with each shipment. NUSS software can also be downloaded from Pressure Systems' website at http://www.pressuresystems.com/netscanner_software.html.

2.2 Computer Preparation

The PC must have its IP address and subnet mask set to be compatible with the addressing existing in the module. For a new module this will be the factory default. The IP address of the host PC must be set to: 200.200.1.1.

Follow this procedure to set the IP address and subnet mask.

1. Click on the Network Neighborhood icon and then right-click to reveal a drop menu. Select **Properties**. The screen that follows should appear.

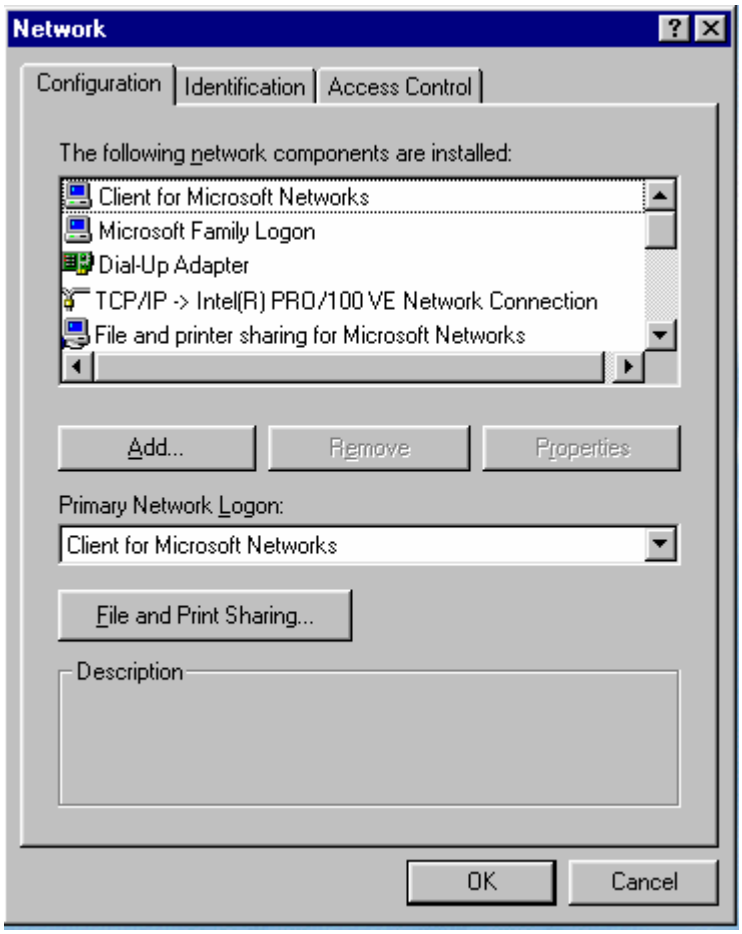


Figure 2-1: Network Dialog Box

2. Using the scroll bar, scroll down to the TCP/IP entry for the Ethernet interface in use by the PC. Select this item and then click the **Properties** box. The screen that follows should appear.

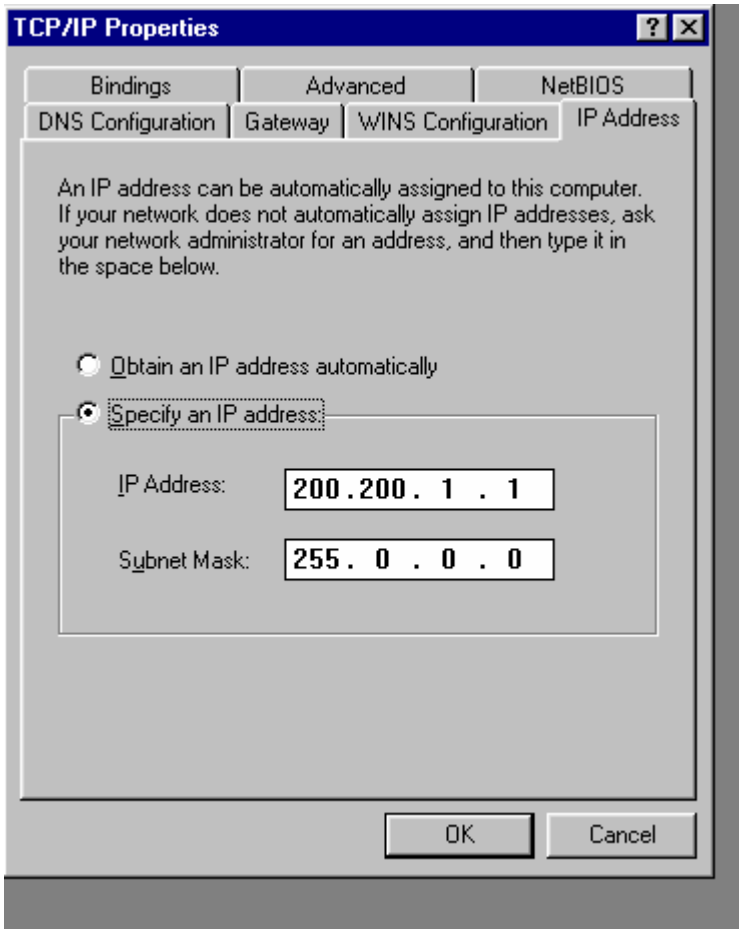

TCP/IP Properties
Bindings
Advanced
NetBIOS
DNS Configuration
Gateway
WINS Configuration
IP Address
An IP address can be automatically assigned to this computer. If your network does not automatically assign IP addresses, ask your network administrator for an address, and then type it in the space below.
Obtain an IP address automatically
Specify an IP address:
IP Address:
200.200. 1 . 1
Subnet Mask:
255. 0 . 0 . 0
OK
Cancel

Figure 2-2: TCP/IP Properties Dialog Box

Note

Be sure to record the subnet mask addressing settings prior to changing them to the values shown to facilitate restoring the computer to its previous network configuration.

3. Move the cursor to the IP address and change it to the one shown. Do this also for the subnet mask. Click OK.

4. In order to put the change in effect, re-boot the PC if prompted to do so.

2.3 Module Connection

The module must now be connected to the PC via the 9IFC interface. Connect the cables as shown in Cable Connections.

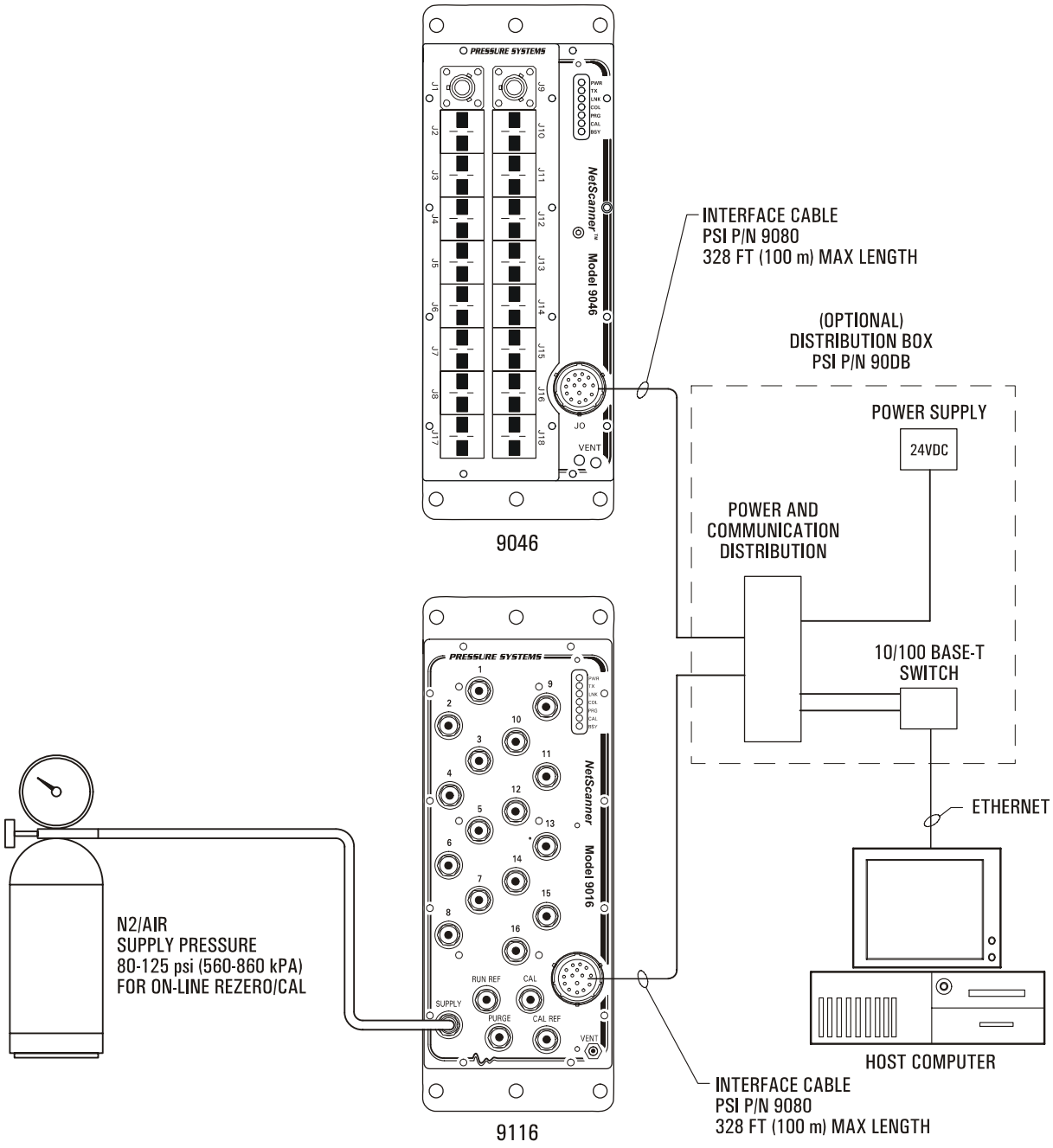


Figure 2-3: Cable Connections

After supplying power to the module, observe the front panel for the following indications:

- PWR LED should remain ON. If this LED is not ON, all other LED's will also likely be OFF. Check the PSI 90DB, 9IFC or user-provided power supply to ensure the proper voltage (18-36 VDC) is being provided. Also verify that the power pins in the module interface cable are wired as described in this manual, Appendix A.
- COL LED should remain OFF.
- TX LED should remain OFF. Note that any activity of the TX LED during the power-up sequence is an indication that the RARP/BOOTP protocol is enabled. This will typically occur following the initial BSY LED cycle and continue until an appropriate RARP reply is received.
- LNK LED should remain ON. This LED indicates proper connection to an Ethernet hub or switch. If this LED is OFF, verify that the module is properly connected to the communications hub or switch. Verify proper power is applied to the hub. Also try connecting the module cable to a different port of the hub. Note that most hubs have similar link LED's to indicate proper connection to the hub itself. If present, verify that the hub link LED for the scanner and the host computer are both active. If the hub is functioning correctly, verify that the communications pins in the module interface cable are wired as described in this manual.
- CAL LED should remain OFF.
- PRG LED should remain OFF.
- BSY LED should blink 8-16 times, and then remain on for about 5 seconds. After that it will blink at about a 100 Hz rate, which will cause the LED to appear to be on, but dimmer than the other LEDs that are ON.

2.4 IP Address Assignment

The module IP address assignment can only be changed if your computer is able to communicate with the module:

1. Use the PSI program NUSS to determine the current module IP address. Change the module IP address to one compatible with your system host computer. Depending on your operating system, it may be necessary to change your Host IP Address temporarily to allow connection to the module.

2. Upon completion of addressing, change the Host computer IP address back to its previous value, and verify that the Host can connect to the Model 9046 module using NUSS.

The module and PC are now ready to communicate. Follow this procedure to assign an IP address:

1. From the start menu select NUSS. The Network Status Screen should appear. Note that a "Querying Network" message briefly appears, then the Nodes On Network Box will display the node map (the types of modules connected and the serial number of each module).

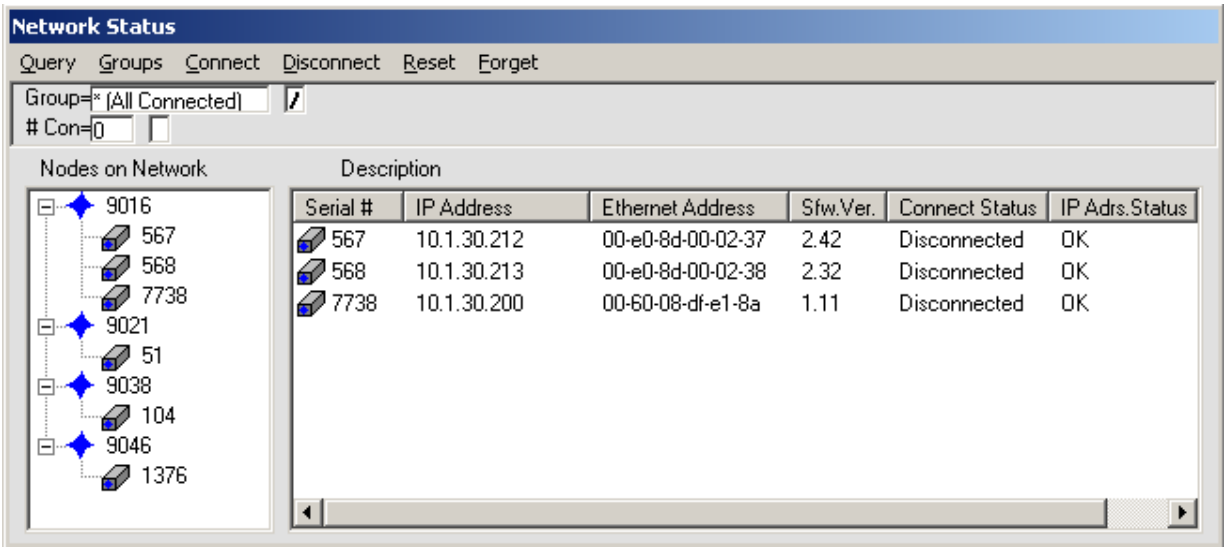


Figure 2-4: Network Status Screen

2. Note the IP address. The serial number should match the label on the unit. If the module in question is a new unit, the IP address should be the factory default.

3. Connect to the module. (When a connection has been established, the top of the module icon will turn yellow.) Follow these steps:

   a. Left-Click on the module's icon in the Node Map.

   b. Right-Click on the module's icon (in either the Node Map or the Description Box).

   c. Click on "Connect" in the pulldown menu that appears.

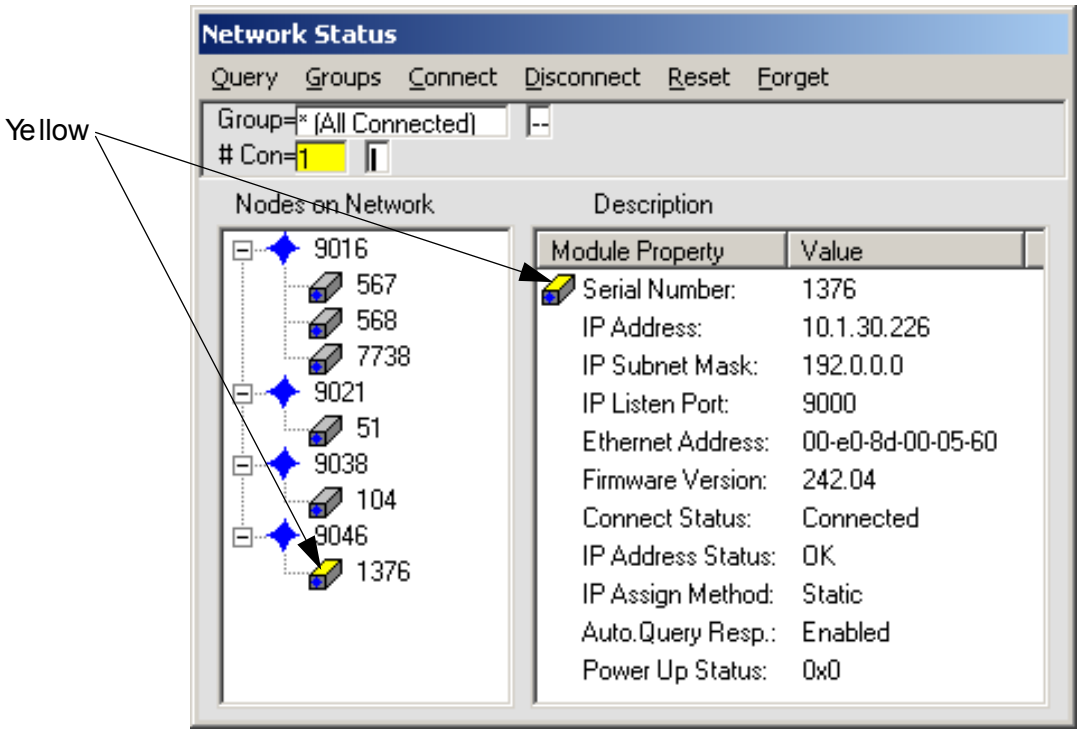


Figure 2-5: Network Status (after connection)

Note

If the module's icon does not turn yellow, the module did not connect.

Possible causes for a failure to connect are:

- There is an IP address incompatibility between the host computer and the module.
- There are two modules with the same address.

4. Open the Network Options (Single Module) Dialog Box:
   a. Right-click on the module's icon (in the Description Box).
   b. Click on "Configure".
   c. Click on "Network Options".

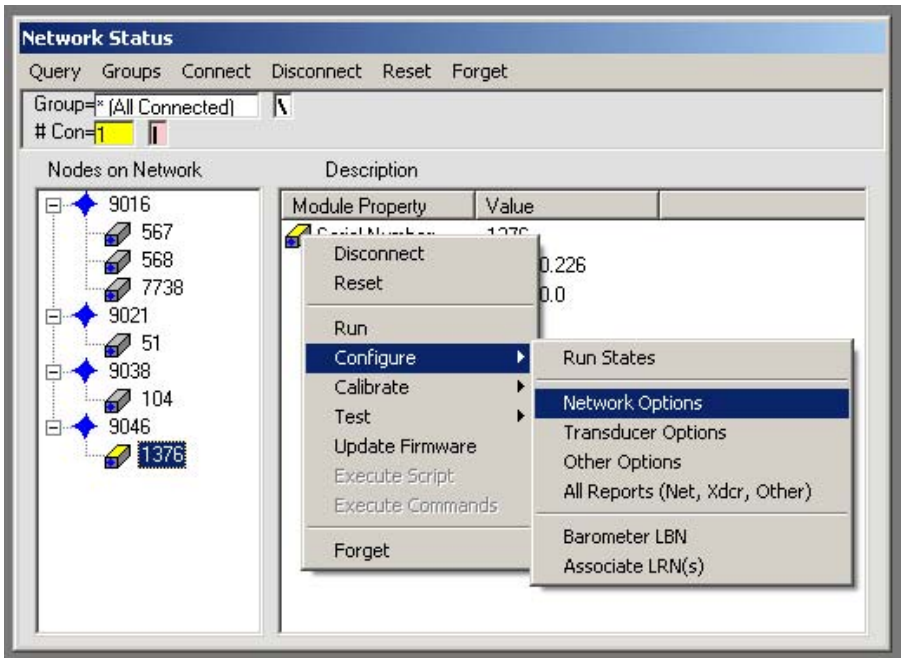


Figure 2-6: Configure>Network Options Pulldown Menu

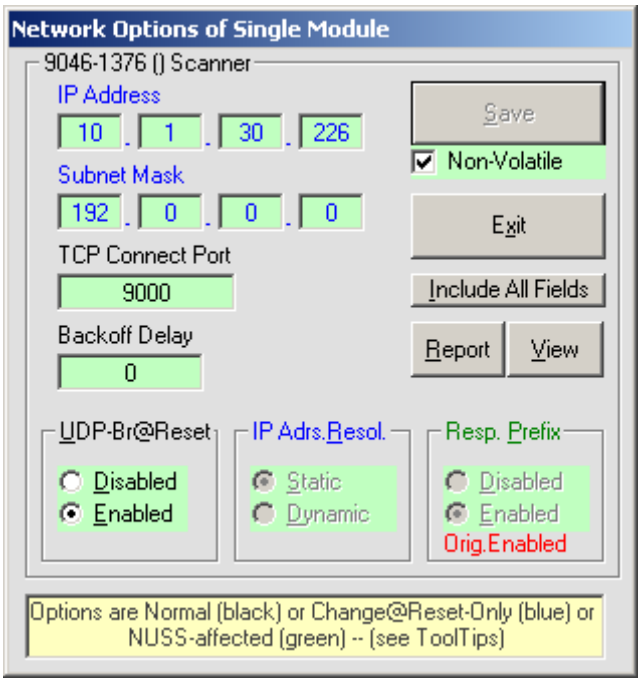


Figure 2-7: Network Options Dialog Box

5. Set the subnet mask and IP address if they are not correct:

   - IP Address - use the address called for by the system.
   - Set the subnet mask to **255.255.255.0**
   - Verify that the TCP Connect Port (Listen Port) is set to **9000**.

6. Click on the **Save** button. The window at the bottom of the Network Options Dialog Box will indicate whether the operation was successful. If the information is not received, check to see that the address was entered correctly.

Note

Clicking "Save" puts the new addresses in the module's EEPROM, but they will not be active until the module is re-booted or the power is removed and re-applied.

7. Click the Exit button. Right-click on the module icon, and select Disconnect.

8. Remove power from the Model 9046, wait 5 seconds and re-apply.

Note

The computer and the module now will have incompatible IP addresses. You will have to change the IP address to one that is compatible. Exit the NUSS program, and change the computer IP. Re-start NUSS. Observe the IP address when it is presented and verify it.

9. This completes the addressing procedure. Label the module with the new IP address.

2.5 Set Sensor Type

1. Confirm that the correct SCM is installed and jumpered correctly:

    a. For Non-Universal SCMs:

    Confirm that each of the 16 hardware channels contains the correct SCM for the intended application. For isolated thermocouple applications, a SCM is required.

    b. For Universal SCMs:

Confirm that each of the 16 SCMs is configured for the intended application. For thermocouple applications, a jumper must be installed in the small sockets on the top of the SCM. (See Figure 1-2).

Signal Conditioning Module (SCM).) For all other applications, no jumper is required.

2. Using Table 1 (on the next page), determine the value for each desired sensor type.

3. Using NUSS configure each channel using a command of the format: **vxxx51 yy**, where **xxx** ranges from 001 through 010 **HEX** for channels 1 through 16, **yy** is the sensor type value determined from Table 1.

Examples:

- **v00a51 18** will set channel 10 to a type T thermocouple.
- **V00951 60** will set channel 9 to an RTD with an ANSI 385 curve.

4. Send a **w07** command to write these values to non-volatile ROM.

Note

When the 'v' command is issued to set the sensor type for a channel, the zero and offset coefficients for the channel are re-initialized to the default values of 0.0 and 1.0 respectively.

Table 1: Sensor Type Values

Sensor Type	Value Assigned
Volts	0
Pressure (volts)	1
Type B Thermocouple	11
Type E Thermocouple	12
Type J Thermocouple	13
Type K Thermocouple	14
Type N Thermocouple	15
Type R Thermocouple	16
Type S Thermocouple	17
Type T Thermocouple	18
Resistance Range 220 W	30
Resistance Range 440 W	31
Resistance Range 990 W	32
Resistance Range 19.8 kW	33
2.25 k Thermistor	40
5 k Thermistor	41
10 k Thermistor	42
RTD385	60
RTD7990	65

2.6 Read Sensor Type

Each channel type can be read using a command of the form: **uxxx51**. where **xxx** = the channel # in **HEX**.

Example:

- **u00a51** will return 18, indicating the channel is configured for a type T thermocouple.

2.7 Thermocouple Reference Temperature Alarms

This feature makes use of existing commands to identify any thermocouple channel with a cold junction reference temperature that deviates from the average of all cold junction reference temperatures by more than a user defined amount (factory default = 10 °C). Related commands are as follows:

Command:	**q0c**
Action:	Query thermocouple reference temperature alarms.

Command:	**w1901**
Action:	Set thermocouple reference temperature alarms threshold.

Command:	**q0e**
Action:	Query thermocouple reference temperature alarms threshold.

Command:	**c05 0002**
Action:	Enable thermocouple reference temperature alarms in autonomous data stream.

2.8 Set Temperature Units

Set temperature units for all temperature channels. The 'v' command is used with array index aa = 11 (hex) and new coefficient index = 0b (hex).

Command:	**v0110b dd** **dd** = 00 for degrees C, **dd** = 01 for Fahrenheit.	
Response:	Normal:	A
	Error:	N08

2.9 Read Temperature Units

Read temperature units for all temperature channels. 'u' command is used with array index aa = 11 (hex) and new coefficient index = 0b (hex).

Command:	**u0110b**	
Response:	Normal:	00 = Degrees C, 01 = Fahrenheit
	Error:	N08

2.10 Write EU Filter Parameter

A software filter stage can be configured individually for each channel using the **v** command with coefficient index number 52 hex to write the n value (max = 100) for filtering EU data (EU_delivered = EU_old + EU_new/n - EU_old/n). The factory default value is n = 1 (filter turned off):

Note

n values greater than 15 – 20 will slow down the step response below what is normally used for thermocouples. Values greater than 10 should be evaluated on a case by case basis. EU values are returned 10 times / second regardless of filter n value.

The following example sets channel 1 EU filter parameter to 10:

Command:	**v00152 10**	
Response:	Normal:	A
	Error:	N08

2.11 Read EU Filter Parameter

The EU filter parameter can be read for an individual channel using the 'u' command with coefficient index number 52 hex. The following example reads the EU filter parameter for channel 1.

Command:	**u00152**	
Response:	Normal:	10 (filter value n = 10) Normal
	Error:	N08

3.0 Calibration

Calibration consists of offset correction.

Implement offset correction (automatically calculate and set offset coefficient) on a per channel basis using the 'h' command. Accurately measure the temperature of the select thermocouple sensor for use in the offset correction calibration. When input to the 'h' command, the appropriate offset coefficient is calculated and applied to the selected sensor channel. This value is initially stored in volatile memory.

The offset coefficient is manually accessible via the 'v' and 'u' commands with coefficient index 00. The w07 command commits all of the aforementioned parameters to nonvolatile memory.

Note

h commands use bit-position format of addressing as follows:

USB 0000 0000 0000 1000 LSB would represent channel 4, and would be expressed in the h command as 0008.

Channel 16 & 6 would be represented as USB 1000 0000 0010 0000 LSB or in an h command as 8020.

Note

When the 'v' command is issued to set the sensor type for a channel, the zero and offset coefficients for the channel are re-initialized to the default values of 0.0 and 1.0 respectively.

Examples:

Correct the offset in channel 1 with channel 1 at 100 degree C :

Command:	**h0001 100.0**
Response:	0.294167 (offset coefficient returned)

Manually set channel 1offset coefficient to 0.4:

Command:	**v00100 0.4**	
Response:	Normal :	A
	Error:	N08

Read channel 1 offset coefficient:

Command:	**u00100**	
Response:	Normal :	0.4 (offset coefficient = 0.4)
	Error:	N08

Store coefficients to FLASH PROM

Command:	**w08**
Response:	A

4.0 Firmware

4.1 Description of Operation

Table 1 shows the different sensor types available along with the gain settings and engineering units for each type. Setting channel sensor type automatically causes the gain for that channel to be set according to Table1.

Sensor Type	**Gain Setting Va, Vb**	**Engineering Units**
Volts	1, 1	Volts
Thermocouple, Types: B, E, J, K, N, R, S, T	90, 1	Degrees C (F)
Resistance Range 220 Ω Range 440 Ω Range 990 Ω Range 19.8 kΩ	 90, 90 45, 45 20, 20 1, 1	Ohms
Thermistor 2.25 k, 5 k, 10 k	 1, 1	Degrees C (F)
Platinum RTD (100 Ω at 0 °C + 0.385 Ω/°C)	90, 90	Degrees C (F)
MIL-T-7990B RTD	90, 90	Degrees C (F)

The two voltages for each channel are read during each pass through the data acquisition task. Default number of averages is 64 and default thermal updates is 15 seconds. Va is the transducer voltage. Vb is the voltage of the cold junction transducer. These voltages are used to calculate engineering units for each channel as follows:

Voltage

If a channel is configured for straight voltage then no additional EU conversion is needed. Va is returned as transducer voltage. Vb is returned as cold junction temperature.

Thermocouple

Va is the thermocouple voltage. Vb is the cold junction reference voltage and is saved to be used in cold junction compensation calculations. Cold junction compensation calculations occur at the thermal update interval for the module (default = 15 seconds). Thermocouple temperature in degrees C is returned. If an erroneous temperature value is calculated or no thermocouple is attached to the channel, the value 99999 is returned. Cold junction reference temperature in degrees C is returned. If Vb is out of range, the value 99999 is returned. If the calculated cold junction temperature is above 70 °C, 88888 is returned. If the calculated cold junction temperature is below –35 °C, –88888 is returned. Cold junction reference temperature is only calculated at the thermal update rate for the module.

Resistance

Va is the load voltage. Vb is the source voltage used to calculate the load current. Load resistance in ohms is returned. If an erroneous resistance value is calculated, the value 10 M is returned.

Thermistor

Va and Vb are used to calculate load resistance as described for resistance measurements above. This resistance value is used to calculate thermistor temperature in degrees C. Thermistor temperature in degrees C is returned. If an erroneous temperature value is calculated, the value 88888 is returned. If a temperature value above 200 °C is calculated, 99999 is returned. If a temperature value below –80 °C is calculated, -99999 is returned.

RTD

Va and Vb are used to calculate load resistance as described for resistance measurements above. This resistance value is used to calculate RTD temperature in degrees C. RTD temperature in degrees C is returned. If an erroneous temperature value is calculated, the value 88888 is returned. If a temperature value above 850 °C is calculated, 99999 is returned. If a temperature value below –200 °C is calculated, -99999 is returned.

4.2 Updating the Firmware

Caution

Special precautions must be taken when updating NetScanner™ firmware. If care is not taken, the module program memory can be corrupted rendering the module inoperative until the program memory chip is physically replaced. Never download firmware not made specifically for the Model 9046.

This page intentionally blank.

5.0 Command Set

Command Quick Reference

Type	Operation Code	Command Function
TCP/IP Commands	A	No Operation
	B	Reset
	V	Read Primary Transducer Volts
	a	Read Primary Transducer A/D Counts
	b	Read Primary Transducer E. U. Data at High Speed
	c	Define/Control Autonomous Host Streams (6 sub-commands)
	h	Calculate and Set Offsets
	m	Read UTR/Other A/D Counts
	n	Read UTR/Other Volts
	q	Read Module Status
	r	Read Primary Transducer E. U. Data
	t	Read UTR/Other E. U. Data
	u	Read Internal Coefficients
	v	Download/Set Internal Coefficients
	w	Set/Do Operating Options/Functions
UDP/IP	psi9000	Query Network

Commands	psireboot	Reboot Specified Module
	psirarp	Change Specified Module's IP Address Resolution Method & Reboot

Note

The commands in this section are quite detailed and involved to anyone wishing to only use the 9046 module. The commands need be understood fully only by persons wishing to program software that talks to the module. If you wish to avoid learning these commands, use the PSI Windows application program called NUSS (NetScanner Unified Startup Software) instead. Versions 1.1.11 or later recognize any 9046 module (with version 2.36 firmware or later), and provide simple point-and-click methods of executing most of these commands. In many cases, it configures complex and useful operating functions by automatically selecting default command sequences to operate the module. However, it will also allow you to manually execute each command (singly or in prepared script files). At the end of each command description below, there are notes similar to this one, indicating how NUSS utilizes (or avoids using) that particular command.

5.1 Summary Information About All Commands

The command descriptions that follow are listed in the same order as the Quick Reference table above. All TCP/IP commands are listed first (in alphabetical order: first by upper-case letters, then by lower-case letters). A unique TCP "connection" must be established between the host computer and the Model 9046 module before such commands can be sent to the module. Finally, there are a few special UDP/IP commands listed that can be sent to a Model 9046 module without a valid TCP "connection" having to be established. These commands can be sent even if the module currently has no TCP address assigned to it.

The 9046 module can actually make *two* (2) separate measurements for *each* of its 16 channels. The *first* is called the *Primary Transducer* datum. For modules configured as *Thermocouple* (TC), RTD, or *Thermistor* Sensor Types this datum will be the *temperature* measurement from a particular channel's *active external junction or transducer*. This datum is available in engineering units (E. U.) of *Centigrade* or *Fahrenheit degrees* (ºC or ºF), *raw A/D counts* (-32768 to +32767), or *volts* (-5 to +5). For *Resistance* Sensor Types this E. U. datum is in *ohms* instead of degrees, and is available in raw counts or volts, too. The *second* data measurement available from each channel is called the *Other Transducer* datum. For *Thermocouple* (TC) Sensor Types, this datum measures the Uniform Temperature Reference (UTR) junction just inside the module. Its E. U. *temperature* value is also in ºC or ºF. For all other Sensor Types this *other* datum measures the *Source Voltage*: an intermediate measurement used in calculating the Primary Transducer value. Its E. U. value is always expressed in volts. This *other* datum is also available in raw A/D counts and volts.

Whether any E. U. Temperature datum value (primary or other) is returned in ºC or ºF, in a command response or stream, is a configuration option of any Model 9046 module (see the '**v0110b**' command below). However, temperature values in some setup commands and responses require a fixed ºC value (e.g., the **UTR Avg. Deviation Set Point** which are written and read (respectively) with the '**w1901**' and '**q0e**' commands below)

The various *parameters* of most commands (following the single ASCII character *operation code*) are concise and efficient (but somewhat arcane). Such parameters are mostly variable-length *hexadecimal* digit fields. Most hexadecimal fields (1 or more hex digits in width) represent the values of simple positive integers, which may represent format choices, modes, or other similar numerical quantities. However, some represent *bit maps;* which are usually four hex digits (representing 16 discrete internal bits, numbered 16-1, left-to-right). Usually each bit represents some on/off (1/0) state, which could be the contribution of a particular channel number (e.g., 16-1), or the presence of a particular type of data grouping in a stream. Other *decimal or hexadecimal* digit fields may optionally appear following the first *space* character in some commands. Please note that all command operation codes are case-sensitive letters (i.e., command 'A' and command 'a' are different commands). However, all "alpha numbers" (a-f or A-F) that are valid of hexadecimal digits are not case sensitive (i.e., 12e4f and 12E4F and 12e4F are the same hex number).

Command responses returned by the Model 9046 module often contain only a simple **A** (Ack) or **N** (Nak) character, indicating the success or failure of the command sent to it. These responses may be followed, or replaced, by one or more hexadecimal or decimal values, as indicated in the particular command description below. If the response is "pure data", requested by some

"Read Data" command, the first character will not contain an **A** or **N** character, and each datum may be in a format specified by a parameter of the command itself, or be implied by the particular command operation code. One or more data fields may appear in a "pure data" response, and these may be contiguous or have separator characters.

Streams are special transmissions output by a Model 9046 module that, once configured and started by special forms of the '**c**' command, will be emitted autonomously (i.e., continue to be output at some rate repetitively, on its own). Subsequent '**c**' commands may stop such streams - if the streams themselves are not programmed to eventually expire on their own. A maximum of three separate concurrent streams can be programmed to be output by any one Model 9046 module. These streams are very useful for acquiring data (at up to three separate concurrent rates) with minimal intrusion by the host.

Most TCP/IP commands, that return immediate data in their response (or cause data to be eventually returned via autonomous data streams), have a *format code* (a number between 0 and 8) that establishes the format of all the data, both sent or returned. The first few format codes (0, 1, 2, and 5) select various "human readable" data. These are ASCII numerical strings with individual data values separated by space characters. Such formats require conversion between the host command format and the native internal module data format (always 32-bit integers or 32-bit IEEE single floating point (big endian) quantities). Normally, the host command or response contains numbers with a decimal (with sign and decimal point) or hexadecimal (dump) representation. The last two format codes (7 and 8) send or return pure binary data (32-bit IEEE Floating Single or Integer, big-endian or little-endian) that only a computer can read. Such formats are the most efficient for transferring data between host computer and module firmware since little conversion is required at each end. When more than one datum of this type appears in a command, response, or stream, the individual data values follow one another (4 bytes/32-bits at a time) without any separators between them. Each command description below, with format choices, has a small table showing the valid formats allowed. A few commands have a fixed format for sending or returning data, which is stated explicitly in its command description.

The Model 9046 has several *special datum values* it returns when a particular E. U. datum is out of range, or was calculated incorrectly. Such values vary with the Sensor type of any channel. All Thermocouple (TC) channels have an "open circuit" detection feature that indicates when a valid thermocouple is unplugged from its channel. Any TC *primary* E. U. datum will be set to approximately 99999 to indicate this condition. The same value is returned for its *other* UTR junction E. U. temperature if the internal measured junction voltage is "out-of-range". Additionally, any UTR junction E. U. temperature above 55 ºC returns the value +88888; any below -35ºC returns a negative value (-88888). Any "out-of-range" Resistance (R) channel returns the E. U. datum 10,000,000 ohms. Any erroneous temperature calculation for an RTD or Thermistor channel returns the E. U. datum 88888. The E. U. value +99999 is returned for RTD temperature values above 850 ºC; any below -200 ºC returns a negative value (-99999). The E. U. value 99999 is returned for Thermistor temperature values above 200 ºC; any below -80 ºC returns a negative value (-99999). When checking for these special E. U. values use an approximate value for comparison (e.g., for 99999 check for greater than (>) 99998.0), since the actual value can sometimes have a fractional part (e.g., 99998.1234). For the special '**b**' command only, these values may be scaled down (e.g., 99998.1 may be 999.981).

The individual description of each command follows immediately. Each command has a functional title (and operation code), followed by a short text description. Next, there is a formal definition of the command format and its response, followed by a more detailed text discussion. The description ends with examples and a NUSS Note (an indication how the NUSS application program of PSI utilizes this command, or provides an easier-to-operate function that is a viable alternative to sending commands manually).

5.2 No Operation (Command 'A')

Purpose: This command has no internal module effect. It is used as a simple method to verify proper communications to the Model 9046 module.

Command	"A" 'A' is the operation code.
Response	"A" or "Nxx" 'A' for acknowledge; 'N' for negative acknowledge, error code xx.

Description: This command is generally used as a simple 'NOP' mechanism to verify proper communications with a module.

Example:

Command module to acknowledge module power on:

"**A**"

Read following response:

"**A**"

Note

NUSS executes this command when you use its 'Connect' function to open a unique TCP Socket for communicating with the module. It uses the command to determine if the module is currently adding a "size prefix" (or not) to each command-response or stream. NUSS always operates the module with this "prefix" enabled - and restores the module to its original "prefix" state when it exits (and disconnects each "connected" module from its unique TCP Socket).

5.3 Reset (Command 'B')

Purpose: Instructs the module to reset internal operating parameters, and to set all internal control variables to their default "reset" state (see description below). The current TCP/IP socket connection will remain open. Execution after a power off/on cycle is optional (unnecessary).

Command	"B" 'B' is the operation code.
Response	"A" or "Nxx" 'A' for acknowledge; 'N' for negative acknowledge, error code xx.

Description: The module returns to the following "reset" states if this command is executed:

- The Offset calibration adjustment coefficient for each transducer is set to the last value stored in flash memory.
- The Gain calibration adjustment coefficient for each transducer is set to the last value stored in flash memory.
- Number of Samples for Data Averaging is set to last value stored in flash memory (factory default = 64).
- Any autonomous host data delivery streams defined by 'c' sub-commands are reset (undefined).

Example:

Command module to reset "power-up" defaults:

"**B**"

Read following response:

"**A**"

Note

NUSS does not use this command normally. You may execute it manually via the 'Execute Commands' or 'Execute Script' functions of NUSS - though this should be avoided if NUSS is currently operating the module via other NUSS functions.

5.4 Read Primary Transducer Volts (Command 'V')

Purpose:

Returns the most recently acquired *primary transducer (i.e., temperature/ resistance)* raw data for the specified channels. These data are averaged signed A/D counts (in the range -32768 to +32767). This simple data bypasses using the transducer's E. U. conversion coefficients (c0..c3) and the *calibration adjustment* coefficients (offset and gain). Each (non-binary) datum returned in the response will be in the specified high-precision data format. (NOTE: the formula for converting A/D counts to volts is: *Volts = A/D Counts * 5/32768*) **This command is intended for advanced users only and is not required for normal operation.**

Command	"**Vppppf**" 'V' is the operation code 'pppp' position field (channel selection bit map) 'f' is the format field
Response	"**dddd.. [dddd]**" 'dddd' are the data fields, each with a leading space (except f =7 or 8).

Description: The 4-character hex position field (pppp) specifies a 16-bit binary bit-map, with each bit (set to 1) to specify a particular channel number (16-1, left-to-right).

The 1-character *format* field (*f)* specifies the format of each *data* field (*dddd*) that will be returned in the requested response. The first datum returned in the response will be for the highest channel number requested. Each (non-binary) datum will be preceded by a space character. Some formats may not be applicable to the specific type of data being requested. Valid formats are shown in the following table:

f	converts each internal response datum value from:	max.char.
0	single binary float to 7-10-digit signed decimal " [-xxx]x.xxxxxx"	13
1	single binary float to 8-digit hex integer " xxxxxxxx"	9
2	double binary float to 16-digit hex integer " xxxxxxxxxxxxxxxx"	17
5	single binary float to long integer (EU*1000) then to 8-digit hex integer	9
7	single binary float to single binary float (big endian: msb first)	4

8	single binary float to single binary float (little endian: lsb first)	4

Example:

Command module to return decimal voltage data for Primary Transducer channels 1, 5, 9, and 13:

"**V11110**"

Response contains data for channels 13, 9, 5, and 1 (left to right):

" 4.999999 -4.989500 0.005390 2.500001"

Note

NUSS does not use this command normally. You may execute it manually via the 'Execute Commands' or 'Execute Script' functions of NUSS - though this should be avoided if NUSS is currently operating the module via other NUSS functions. NUSS normally reads all the data it needs from a module by configuring autonomous streams (see the 'c' command).

5.5 Read Primary Transducer A/D Counts (Command 'a')

Purpose:

Returns the most recently acquired *primary transducer (i.e., temperature/ resistance)* raw data for the specified channels. These data are averaged signed A/D counts (in the range -32768 to +32767). This simple data bypasses using the transducer's E. U. conversion coefficients (c0..c3) and the *calibration adjustment* coefficients (offset and gain). Each (non-binary) datum returned in the response will be in the specified high-precision data format. (NOTE: the formula for converting A/D counts to volts is: *Volts = A/D Counts * 5/32768*) **This command is intended for advanced users only and is not required for normal operation.**

Command	"appppf" 'a' is the operation code 'pppp' position field (channel selection bit map) 'f' is the format field
Response	" dddd.. dddd" 'dddd' are the data fields, each with leading space (except f = 7 or 8).

Description: The 4-character hex position field (pppp) specifies a 16-bit binary bit-map, with each bit (set to 1) to specify a particular channel number (16-1, left-to-right).

The 1-character *format* field (*f)* specifies the format of each *data* field (*dddd*) that will be returned in the requested response. The first datum returned in the response will be for the highest channel number requested. Each (non-binary) datum will be preceded by a space character. Some formats may not be applicable to the specific type of data being requested. Valid formats are shown in the following table:

f	converts each internal response datum value from:	max.char.
0	single binary float to 7-10-digit signed decimal " [-xxx]x.xxxxxx"	13
1	single binary float to 8-digit hex integer " xxxxxxxx"	9
2	double binary float to 16-digit hex integer " xxxxxxxxxxxxxxxx"	17
5	single binary float to long integer (EU*1000) then to 8-digit hex integer	9

7	single binary float to single binary float (big endian: msb first)	4
8	single binary float to single binary float (little endian: lsb first)	4

Example:

Command module to return decimal raw "Temperature/Resistance" A/D counts data for Primary Transducer channels 1, 5, 9, and 13:

"**a11110**"

Response contains data for channels 13, 9, 5, and 1 (left to right):

" **32767.000000 -32700.000000 10.000000 16385.000000**"

Please note that channel 13 is saturated at +full scale and channel 9 is almost saturated at -full scale. Channel 5 reads near zero and channel 1 is about 1/2 +full-scale.

NUSS does not use this command normally. You may execute it manually via the 'Execute Commands' or 'Execute Script' functions of NUSS - though this should be avoided if NUSS is currently operating the module via other NUSS functions. NUSS normally reads all the data it needs from a module by configuring autonomous streams (see the 'c' command).

5.6 Read Primary Transducer E. U. Data at High Speed (Command 'b')

Purpose:

Returns the most recent Primary Transducer E. U. data from *all channels* of the module as fast as possible. Data is returned directly in its native internal (IEEE single-precision float) binary form (i.e., as for format code 7 in other commands). It is used as a *faster alternative* to the ***Read High-Precision Data*** (**'r'**) command, since **'b'** does not have to parse any parameters, nor does it have to transform or encode the internal data into any other format when the response is generated.

Command	**"b"** 'b' is the operation code
Response	**aaaabbbbcccc..pppp** Note that this response if not in quotes, because it is a stream of binary bytes. Each 4-byte datum (e.g, aaaa) is a non-human readable big-endian value (format 7) representing an IEEE single-precision internal float value.

Description: Returns primary E.U. data for all of the module's channels in order highest channel number to lowest channel number. Thus channel #16 will always be the first 4-byte (32-bit binary, big-endian, IEEE floating-point) value ('*aaaa*') sent in the response. Any *special datum values* (e.g., 99999 or 88888) used to indicate out-of-range or erroneous calculations or physical connections are re-scaled to lower values (e.g., 999.99 or 888.88) for this command only.

Example:

Command module to return all its primary transducer E. U. data as fast possible:

"b"

Data from the most recent scan of all the module's channels are returned in pure binary form, 4-bytes per channel (big endian) of each internal IEEE float:

aaaabbbbcccc .. pppp

NUSS does not use this command normally. You may execute it manually via the 'Execute Commands' or 'Execute Script' functions of NUSS - though this should be avoided if NUSS is currently operating the module via other NUSS functions. NUSS normally reads all the data it needs from a module by configuring autonomous streams (see the 'c' command).

5.7 Define/Control Autonomous Host Streams (Command 'c')

Purpose: Defines and controls the autonomous delivery of up to three concurrent high-speed *autonomous data streams* to the host computer. Such *data streams* may be delivered "continuously" (i.e., unbounded, until a command explicitly stops them), or be delivered in a "limited" amount (i.e., bounded, by a pre-specified fixed number of data scans to be sent). Each scan delivered may be synchronized by a user-supplied "hardware trigger" or each scan may be delivered periodically as synchronized by an internal software clock. These concurrent host streams are an alternate method of acquiring/delivering data rather than the host using the many other special purpose "read" commands ('***V***', '**a**', '**b**', '**r**', '**t**', '**m**', and '**n**') for reading various data values in various formats.

Host data streams, once activated in a module, normally deliver a sequence of TCP/IP (optionally UDP/IP) data scans *autonomously* to the host (i.e., without the host sending any particular command to the module to request each scan).

Note

If data streams are defined to occur at high scan rates, then each data scan received by the host must be processed and disposed of in a timely manner. Model 9046 modules are capable of generating autonomous data streams faster than some "slow" hosts (or limited software, including NUSS) can absorb.

Various *sub-commands* (described on the following pages) are used to identify the various definition and control options of the following *general* 'c' command.

Command	"c ii[dddd] ... " 'c' is the operation code ' ii' is a space followed by a sub-command index (augment code). ' dddd' are one or more optional datum fields, each preceded by a space character; These parameters differ for each sub-command. NOTE: all parameters are preceded by a space, including the first hex parameter (ii) - which is unlike most other commands which attach the first parameter(s) to the operation code.
Response	Depends upon particular sub-command (' ii') sent. See below.
Autonomous Packet	Depends upon the particular sub-command (' ii') sent. See below.

Description: The firmware of any module, once fully initialized after Reset (or power-on), continuously scans and converts data for all Primary (Temperature/Resistance) Transducers at the highest possible speed. The result of such scanning is a continuously updated EU *data buffer*, available to three concurrent host data delivery tasks, or available to other standard data acquisition commands in the module. Each host delivery task can grab engineering-unit data values from the EU *data buffer* and deliver them to the host in its own programmable data stream (a sequence of packets (or scans) that autonomously arrive in the host, as long as the host maintains enough socket buffering space to accept them). Other data (e.g., UTR Temperature data for TC sensors and Source Voltage for other sensors) are scanned and buffered at a *slower rate* by a separate Thermal Update task (with its own command configurable scan rate) inside the firmware. These data are also available to the three stream delivery tasks, but should practically be sent only by the slowest stream.

Special augments of this command, called sub-commands (distinguished by the first parameter *ii*) can configure each data stream with the particular channels whose data are delivered, the datum format, the delivery rate, and other characteristics. The sub-command can also start, stop, or undefine a single stream or all defined streams, or change the delivery protocol.

The maximum rate of any one stream's delivery is practically limited to the maximum possible scan and data conversion rate of all the module's channels. Normally, these programmable host streams deliver host data at rates equal to or slower than this natural cycle. For a typical application, the first stream delivers a few channels at a high rate as defined by a hardware trigger. The second stream delivers other channels at a medium rate (some multiple of the trigger), and the third stream can deliver still other channels at a slow rate (a larger multiple of the trigger). In another application, the three streams might all be programmed to deliver all the same channels, but the first stream might deliver Temperature/ Resistance data (i.e., Primary Transducer EU data only) at high speed. The second stream might deliver these same data as counts or volts, but at a slower rate. The third stream might deliver UTR/Volts data (i.e., Other Thermal Update Task data) in all forms (EU, counts, volts) at a very slow rate. Thermal Alarm status may also be returned as a special (bit map) prefix to any configured stream.

Note

NUSS uses various forms of the 'c' command to configure (and control) up to three autonomous data streams in each module. See the similar note after the description of the 'c 00' sub-command below for the default contents of each stream that NUSS configures automatically for each Model 9046 module. When you use the NUSS 'Run' function, you unleash these streams, which are then processed by (and displayed on) the Run form's data fields. Note that only the EU data is displayed by default. Other voltage and A/D count data items are acquired whether or not they are displayed, but can be displayed (on a per channel basis) via several pop-up forms that appear if you click a channel's units or name field. Note: After changing any of module's transducer Sensor Types or any Run form display formats, you must use the 'Forget' function (on module's Context menu) before you can see the results of the change.

Note

When using a module's hardware trigger inputs to synchronize data stream outputs, the frequency of the trigger source should be no more than 200Hz even if the requested output is 100Hz or less. Because the default Scans-Per-Average option is set to 64 (default) in a Model 9046, only a 10Hz scan rate is normally achievable, however, unless the Scans- Per-Average option is reduced.

5.7.1 Command 'c'- Sub-command Index 00: *Configure a Host Delivery Stream*

This sub-command is used to configure the *principle* parameters of each of the three possible concurrent host delivery streams, one at a time. Following this configuration phase, the *particular* stream (1, 2, or 3) or *all* streams (0) may be started, stopped, altered, or undefined, with other sub-commands. This sub-command's format is:

Command	"c 00 st pppp sync per f num" 'c' is the operation code '00' is the sub-command index (ii) for configuration 'st' is the stream id digit (1, 2, or 3) 'pppp' is a 1-4 hex digit position field (channel selection bit map) 'sync' is sync type character (0= hardware trigger or 1= clock) 'per' is the period (if sync=0: # of trigger periods or if sync=1: delay timer period in msec). 'f' is the format of each acquired datum in stream 'num' is the number of scans delivered in the stream (0=unbounded, >0 bounded). NOTE: all parameters are separated by a space character.
Response	"A" or "Nxx" 'A' for acknowledge; 'N' for negative acknowledge, error code xx.
Autonomous Packet	none generated (until Stream Start sub-command)

Description: Configures a particular stream ('***st***') to deliver data scans autonomously to the host, with each scan containing selected acquired data for the channels specified. The channels are specified by a standard 16-bit *position* field bit map (encoded *pppp* as a 1-4 hex digits). A separate sub-command (*ii*=05) may be used to select which *acquired data groups or prefixes* are included in each stream. By default, only Temperature/Resistance (EU) data group is selected (if *ii*=05 sub-command is never executed for the stream after its initializing 'c 00' sub-command).

The generation of individual scans of a stream may be synchronized with either an external user-supplied hardware trigger or a periodic (1000 Hz) clock interrupt generated inside each module. This choice is made with the *sync type* '***sync***' parameter (a single digit) where: 0 = synchronize with hardware trigger; 1= synchronize with the periodic clock.

When the **hardware trigger** is used to synchronize **stream scans** (*sync* = 0), it is assumed that the user would prefer to also synchronize internal data acquisition cycle. For this reason, when a stream utilizing hardware trigger is started, the module firmware switches out of the free-running continuous data acquisition mode described earlier. Instead, the module waits in an idle mode until a hardware trigger is received to initiate a host stream output. Only on the receipt of that hardware trigger will the module scan and EU convert all averaged scans of all attached channels. Following completion of the acquisition and EU conversion cycle, the module will also deliver the requested data channels to the host in any configured streams. In this manner, users are provided with highly synchronized data acquisition *and* delivery from one or more modules. If a module waits in the idle mode for an extended period of time without receiving a data request, it will periodically initiate its own internal data acquisition cycles to update internal Thermal Scan data and check UTR Thermal Alarms.

When all hardware triggered streams are complete or terminated, an individual module will return to the default mode of continuous scanning and EU conversion. When the **internal software clock** is used to control host stream output rates (*sync*=1), note that 1000 Hz clock frequency variances will result in slightly different timing between modules. Although these differences in timing are slight, they may result in noticeable differences in output timing between modules over a long period of time. If highly synchronized data output is required from multiple modules, a common hardware trigger wired to all modules should be used.

The period '***per***' parameter is a positive decimal integer count (from 0 to 2147483647 (largest 32-bit positive number), specified with 1 to 10 numeric digits as needed), and its meaning depends on the *sync type* '***sync***' parameter described above.

'**sync**	meaning of '**per**'
0	number of hardware trigger periods to wait before sending each scan
1	delay period (in milliseconds) to wait before sending each scan NOTE: Values 0 to 9 will select a minimum 10 millisecond period.

The '***f***' parameter identifies the *format* of each selected acquired datum in each data group of the stream scan, and is a single numeric digit. Valid format codes are listed in the following table:

f	converts each internal response datum value from:	max.char.
0	single binary float to 7-10-digit signed decimal " [-xxx]x.xxxxxx"	13
1	single binary float to 8-digit hex integer " xxxxxxxx"	9
2	double binary float to 16-digit hex integer " xxxxxxxxxxxxxxxx"	17
5	single binary float to long integer (EU*1000) then to 8-digit hex integer	9
7	single binary float to single binary float (big endian: msb first)	4
8	single binary float to single binary float (little endian: lsb first)	4

Note

With the exception of binary formats 7 and 8, all other formats include at least one leading space in each datum delivered in each stream scan.

The *number of stream scans* (*'num'*) parameter is a positive integer count (from 0 to 2147483647, specified with 1 to 10 numeric digits as needed). It sets a finite limit on the number of scans delivered in the host data stream. The value 0 for this parameter requests "continuous" (i.e., unbounded) output scans of the defined host stream.

Example 1:

Configure three (3) separate autonomous host delivery streams, and divide the module's channels between them. Channels (1-4) must be delivered to host as fast as possible, channels 5-8 may be delivered at half that rate, while the remaining channels 9-16 are delivered at half the previous rate. All streams are generated continuously (without bound) and synchronized with the internal clock at 100 msec., 200 msec., and 400 msec. periods, respectively. Data are requested in single precision binary big-endian IEEE float format (*f*=7). This is the module's most efficient "native" format.

"c 00 1 000F 1 100 7 0"
"c 00 2 00F0 1 200 7 0"
"c 00 3 FF00 1 400 7 0"

Read response:

"A"
"A"
"A"

Example 2:

To similarly acquire data at "relative" rates (1, 2, and 4) using a periodic hardware trigger (assumed to also cycle periodically at some practical (adjustable) rate), enter the commands:

"c 00 1 000F 0 1 7 0"
"c 00 2 00F0 0 2 7 0"
"c 00 3 FF00 0 4 7 0"

Read responses:

"A"
"A"
"A"

The type of data (also called the Data Group) delivered by default for each specified channel (when the streams are started) is from the Primary Transducer (i.e., Temperature/ Resistance) in E. U., unless sub-command "05" is subsequently executed (before the stream is started) to select other Data Groups (and/or a UTR Alarm State Prefix) in each stream. The "05" sub-command can also de-select the default Primary Transducer (Temperature/Resistance) data in E. U.

Note

NUSS automatically creates a default Run State 0 for every module that uses its Run function. This run state uses the above sub-command to configure three (3) "unbounded" autonomous data streams that the module will deliver concurrently at various rates to the host:

- Stream 1 contains Primary Transducer E.U. data (Temperature/Resistance) only for every channel and repeats every 1/2 second.
- Stream 2 contain a Thermal Alarm Prefix and Primary Transducer Counts and Volts data for every channel, and repeats every 2 seconds.
- Stream 3 contains Other (UTR/Source Voltage) Transducer data (in E. U., Counts, and Volts) for every channel and repeats every 7.5 seconds.

The Run function will acquire all the data defined in the Run State, but will display only the data in the arriving streams that match the form's currently configured Display requirements (called its Display Set). If a user requires other stream contents or rates, these can be configured (as Run States 1-9) using the NUSS Run State Editor. Any run-state can be assigned to the Run form of each module.

5.7.2 Command 'c'- Sub-command Index 01: *Start Stream*(*s*)

This sub-command is used to start the delivery of any previously configured host stream in a module. If the stream started is "unbounded", then it will be necessary to use the Stop Stream sub-command to stop it. Otherwise, a bounded stream will end automatically if a finite number of scans has been specified for it. This sub-command may also be used to resume a previously stopped host stream that has not transmitted all requested data scans. The sub-command's format is:

Command	"c 01 st" 'c' is the operation code '01' is the sub-command index ('ii') for Start Stream(s) 'st' is the stream id digit (1, 2, or 3, or 0=all streams) NOTE: all parameters are separated by a space character.
Response	"A" or "Nxx" 'A' for acknowledge; 'N' for negative acknowledge, error code xx.
Autonomous Packet	"tssss[pp] [dddd] .. [dddd]" 't' is a 1-byte binary (8-bit) value identifying the stream number (1-3). 'ssss' is a 4-byte binary integer (32-bit, big-endian) scan sequence. number. Optional binary status may follow the sequence number. (See "05" sub-command.) 'pp' is an optional 2-byte binary (16-bit, big-endian) UTR Thermal Alarm Bit Map (one bit per channel 16-1, left-to-right, 1=in alarm, 0=not in alarm) ' dddd' are the acquired datum values for each selected channel of each selected group, in the selected format. Each formatted datum is a 9, 13, or 17 byte ASCII string with one or more leading space characters (except the binary formats (f=7 or 8) which are contiguous 4-byte IEEE floats in big- or little-endian, respectively)

Description: This sub-command starts a particular specified host stream (*st*=1-3), or starts *all* configured host streams with a single command (*s*t=0). Each autonomous host stream scan begins with a 5-byte fixed-format (binary) data header (*tssss*). The first byte (*t*) identifies the host stream, while a 32-bit unsigned binary (big-endian) sequence number (*ssss*) completes the header. This sequence number will start at one (1) for the first scan returned by a stream and increment for each other returned scan of that stream. In the case of an unbounded data stream, the sequence number may overflow the maximum permissible 32-bit integer value. If this occurs, the sequence number value will wrap around to zero (0) following the largest 32-bit (unsigned) value (4294967295) and then continue to increment by one for each returned scan. The sequence number field is intended to provide a mechanism for host software to ensure that host data stream scans are processed or stored in the order in which they were obtained by the

module. Each of the three possible host streams will report its own unique sequence number. Note that if a previously stopped data stream is restarted, the returned sequence numbers will resume with the next number at the point of the stream's termination. The sequence numbers will restart at one (1) only if the streams are reconfigured (with '**c 00**' sub-commands) A bounded stream will terminate once this sequence number equals the requested number of scans for the stream.

For periodic hardware-triggered streams, that are never suspended and resumed after being initially enabled, the sequence number may also serve as a "relative" time stamp if the *period* (in milliseconds) of the hardware trigger is known.

If a special sub-command ('**c 05**') is used to select the Data Group content of a stream, other binary *status* data may immediately follow the binary stream header and *precede* the default Temperature/Resistance EU Data (if selected). Other *special* acquired data groups (per selected channel) may follow or replace the Temperature/Resistance EU Data. Each datum group in each scan will be ordered from highest channel number requested to lowest channel number requested. Each datum (*dddd*) will be output per the format code specified when the stream was configured (by sub-command "**c 00**" or combination of "**c 00**" and "**c 05**").

Example:

Start all the streams configured in the previous example:

"**c 01 0**"

Read response:

"**A**"

Soon after the response is received, the requested data stream scans will begin arriving in the host at a quantity, content, and rate determined by each stream's own particular current configuration (per both the "**00**" and "**05**" sub-commands).

Note

NUSS uses this sub-command to start data acquisition on its Run forms. If the default Run State 0 stream configuration is used, all forms of data from a module are acquired for all channels, at various rates (see NUSS note at end of 'c 00' sub-command) and made available to a Run form's Display and Record functions. This default Run State 0 collection of streams will likely contain too much data for "efficient" recording of just E.U. data. For such requirements, NUSS has a Run State Editor for creating other "custom" Run States (1-9) that can be assigned to a Run form. These can generate more efficient stream configurations for high-speed data recording, including hardware-triggered synchronized scans.

5.7.3 Command 'c'- Sub-command Index 02: *Stop Stream*(*s*)

This sub-command is used to stop (or temporarily suspend) the delivery of any previously started host stream in a module, one at a time or all together, whether the stream was unbounded or bounded. The sub-command's format is:

Command	"**c 02** ***st***" 'c' is the *operation* code '02' is the sub-command index ('*ii*') for ***Stop Stream*** '*st*' is the stream *id* digit (single stream 1, 2, or 3, or 0=*all streams*) **NOTE: all parameters are separated by a *space*.**
Response	"**A**" or "**N**xx" 'A' for *acknowledge*; 'N' for *negative acknowledge, error code xx*.
Autonomous Packet	command stops generation of autonomous scans from the requested stream(s).

Description: This sub-command stops the current "run" of a particular specified host stream (*st*=1-3), or stops the current "run" of "all configured" host streams with a single command (*st*=0).

Any stopped stream may be resumed (i.e., restarted) with the ***Start Stream*** sub-command as long as that stream remains defined in the module and any bounded sequence count has not yet expired. The ***Clear Stream*** sub-command may be used to undefine a stream that you no longer want to be generated by a Start All Streams sub-command. Any unbounded stream or unexpired bounded stream that is restarted continues generating new sequence numbers (i.e., at the count where it left off when stopped). However, the stream must be reconfigured with the ***Configure a Host Delivery Stream*** sub-command (00) before it restarts with sequence count =1. Any expired bounded stream must be reconfigured to restart at all.

It is possible for an outstanding stream to arrive in the host between this command and its response. In that case, the stream can be distinguished from the response by its unique prefix (tssss) where t = 0, 1, 2 binary.

Example:

Stop all the streams configured in the previous example:

"**c 02 0**"

Read response:

"**A**"

Note

NUSS uses this sub-command to stop the acquisition of data on each Run form when that form exits, or when the [Acquire] button on the form is clicked once. Clicking the [Acquire] button again restarts the current Run State's acquisition cycle "completely" by sending another set of stream initialization sub-commands (i.e., 'c 00' and 'c 05' for each configured stream) followed by a Start All Streams ('c 01') command. This more lengthy re-init./re-start sequence insures that each stream is restarted at sequence # 1. The user can also "pause" the streams by clicking any of the green boxes that display the current stream sequence numbers. This causes a Stop All Streams ('c 02') sub-command to be sent, and the green boxes turn yellow to indicate a "pause". Clicking any of the yellow boxes again will "resume" the streams by sending a single Start All Streams sub-command. This causes the boxes to turn green again and resume incrementing the sequence counters where they left off. Of course, any "bounded" stream, that may eventually expire, is not restarted by this simpler "resume" process.

5.7.4 Command 'c'- Sub-command Index 03: *Clear Stream*(*s*)

This sub-command is used to "undefine" any previously configured host stream in a module, one at a time, or all together. The sub-command's format is:

Command	"**c 03** ***st***" 'c' is the *operation* code '03' is the sub-command index ('*ii*') for configuration '*st*' is the stream identifier character (1, 2, or 3 or 0=*all streams*) **NOTE: all parameters are separated by a *space* character**.
Response	"**A**" or "**N**xx" 'A' for *acknowledge*; 'N' for *negative acknowledge, error code xx*.
Autonomous Packet	none generated

Description: This sub-command *clears* (un-defines) the particular specified host stream (*st*=1-3), or un-defines "all configured" host streams with a single command (*st*=0). Once cleared, a stream must be reconfigured before it can be started.

Example:

Stop all the streams configured previously. Then clear (un-define) only stream 3. Finally, resume the remaining defined streams 1 and 2:

"c 02 0"
"c 03 3"
"c 01 0"

Read response:

"A"
"A"
"A"

NUSS uses this sub-command to undefined data specific streams when its Run State Editor form is used to define "custom" stream configurations (called Run States 1-9) and all three streams are not to be generated concurrently. All three streams defined in the default Run State 0 are "unbounded" and always remain defined. You can also define, undefine, and control streams manually inside the NUSS 'Execute Script' function, though streams activation should be avoided in the NUSS 'Execute Commands' function (which is not equipped to display them asynchronously).

5.7.5 Command 'c' - Sub-command Index 04: *Return Stream Information*

This sub-command returns current stream configuration information in its response. Its format is:

Command	"**c 04 *st***" 'c' is the *operation* code '04' is the sub-command index ('*ii*') for configuration '*st*' is the stream identifier character (1, 2, or 3 **only**) **NOTE: all parameters are separated by a *space* character**.
Response	"***st pppp sync per f num pro remport ipaddr bbbb*** " 'st' is the *stream identifier* digit (1,2, or 3) '*pppp*' is a hex *position* field (channel selection bit map) '*sync*' is *sync type character* (0 or 1) '*per*' is the *period* (# trigger periods or delay timer period) '*f*' is the *format* of the data delivered in stream '*num*' is the *number of scans **delivered*** in the stream '*pro'* identifies the protocol used for stream delivery (1=UDP/IP, 0=TCP/IP. This protocol identifier pertains to stream delivery **only**. *'remport'* identifies the remote port number to which each stream delivery is directed in the host. A value of -1 indicates that stream delivery is directed to the same port number the host is using to send commands to the module. *'ipaddr'* identifies the IP address of the host to which the stream delivery is directed. *'bbbb' another position field (data group options bit map) as specified by the "**05**" sub-command.* NOTE: All datum fields separated by a *space* character.
Autonomous Packet	none generated

Description:

This sub-command returns the current configuration information for a particular stream. Returned values are defined the same as the "contributing" parameters of the separate sub-commands: *Configure a Host Delivery Stream* ("**00**"), *Select Protocol* ("**06**"), and *Select Data in a Stream* ("**05**"). Note that the '*num*' field represents the number of scans actually returned so far (= last sequence number returned, or =0 if stream has not yet started.

Example:

Return configuration information for stream 1

"**c 04 1**"

Read response:

"**1 FFFF 0 20 7 32000 1 7002 200.200.200.1 0010**"

The above example shows that stream scans all 16 channels. Data is acquired using hardware trigger with one (1) data scan acquired for every trigger events. Data is returned in binary format 7. 32000 scans have been returned so far. Data is sent using UDP protocol to port 7002 at IP address 200.200.200.1. Temperature/Resistance EU data **only** is returned for the requested channels.

Note

NUSS only uses this sub-command (silently, behind the scenes) if a user-written command script (executed by the NUSS 'Execute Script' function) starts a stream without having fully defined it (with other 'c 00' and 'c05' commands) first.

5.7.6 Command 'c' - Sub-command Index 05: *Select Prefix/Data Groups in a Stream*

This sub-command sets additional options that cause an already defined stream to deliver a specific prefix and/or other groups of data to the host. By default, only the Primary Transducer (i.e., Temperature/ Resistance) E. U. data (and no prefix) are delivered for the *channels* and *format* currently specified by the pre-requisite "**c 00**" sub-command for that stream.

Command	"**c 05** ***st bbbb***" 'c' is the *operation* code '05' is the *sub-command index* ('*ii*') for ***Select Prefix/Data Groups***. '*st*' is the *stream id* digit (1, 2, or 3 only) ' *bbbb*' is a hex option field (bit map) to select which prefix/data group options will be returned in the data stream (see table below) **NOTE: all parameters are separated by a *space* character**.
Response	"**A**" or "**N**xx" 'A' for *acknowledge*; 'N' for *negative acknowledge, error code xx*.

Description: If this sub-command is never executed for a particular stream, then **Primary Transducer (i.e., Temperature/Resistance) EU Data** are delivered (by default) in that stream following the fixed format binary header (*tssss* as described by the "**01**" sub-command). However, this sub-command may also *delete* these default data from a stream (by **not** specifying them) as well as *add other* selected acquired data groups to a stream (by specifying them).

The *bit map values* (shown in the following table) may be added together to specify *all* the actual data groups that will be delivered in each scan of the specified stream. The first two table entries, if their "bits" are specified, will cause two-byte binary (16-bit, big endian) *status* values to be delivered in the stream scan (immediately following the binary stream header). The third table entry, if specified, will cause the **Primary Transducer** (i.e., **Temperature/Resistance) EU Data** to be delivered (next), per the specified pre-requisite **'c 00'** sub-command's format (*f*), and for just the channels specified in that sub-command. The remaining table entries will cause other *special* data groups (i.e., Primary Transducer data in counts or volts, and Other Transducer (UTR Temp./Volts) in EU, A/D counts, or voltage forms) to also be delivered in each stream scan. Each of these *special* data groups is also output in the order of its table entry (within the scan) if its "bit" is specified. Each group will also have a datum per the specified channels, and be in the specified format.

Note

Selecting too many data groups will reduce module performance.

bbbb	(hex) data selected for inclusion in each stream packet
0001	(reserved for Valve Status Prefix (9046 does not use)
0002	Enable **UTR Alarm Status Prefix** (see alarm bit map below)
0010	Enable **Primary Transducer (Temperature/Resistance) EU Data** (default if "05" never executed after "00")
0020	Enable **Primary Transducer (Temperature/Resistance) A/D Counts**
0040	Enable **Primary Transducer (Temperature/Resistance) Voltages**
0080	Enable **UTR Temperature/Other EU Data (ûC)**
0100	Enable **UTR Temperature/Other A/D Counts**
0200	Enable **UTR Temperature/Other Voltages**

Any **UTR Alarm Status** Prefix datum is delivered as a two-byte binary bit map (16-bit, big endian) with each bit representing the status of UTR channel #16 through UTR channel #1 respectively. A bit value of 0 (zero) indicates the UTR is operating within its specified limits. A value of 1 (one) indicates the UTR is outside its specified limits (i.e., "in alarm"). To be "in alarm" the channel's UTR datum value has to deviate from the average of the UTR data values of all the TC channels, by an amount greater than a specified set point (in ûC).

Bit#	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Chan #	**16**	15	14	13	12	11	10	9	8	7	6	5	4	3	2	**1**
Binary	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Hex	8				0				0				1			

The above example indicates that Channels 1 and 16 are "in alarm".

Example:

Configure stream 1 to return UTR Thermal Alarm Status prefix field, and all the UTR Temperature/Other E. U. data. Exclude the Primary Transducer (Temperature/Resistance) E. U. Data:

"**c 05 1 0082**"

Read response:

"**A**"

If or when stream 1 is subsequently enabled, data groups in that stream with the *lowest-bit-numbers (table positions)* selected are delivered first. In this example (*bbbb* = 0082), the **UTR Temperature Status** binary prefix datum would be first, and then all the specified **UTR Temperature/Other EU** data would follow (highest specified channel to lowest specified channel). The standard 5-byte binary prefix (*tssss*) that begins all stream scans would precede this status and data group.(See the Autonomous Packet box in **Start Stream** sub-command ('**c 01**')

Note

NUSS uses this command in its Run State Editor to define the data group content of any streams that generate other than the Primary Transducer E. U. Data group only. There is such a sub-command in the definition file for each Run State 'r' (recorded in a file with the name: "9046-<serial#>rs<r>.ini" in the 'Ini' subfolder of NUSS). A similar file with the name: "9046-<serial#>rs<r>.0" in the same subfolder defines the Data Set of Run State 'r'. This contains the Run form's Display Format, Units, and Precision for each Primary and Other data for each channels tabular display field when Run State 'r' is used by the form). See Appendix F of the NUSS User's Manual for more information on these files.

5.7.7 Command 'c' - Sub-command Index 06: *Select Protocol For Stream Delivery*

Command	"**c 06 *st pro* [*remport* [*ipaddr*]]**" 'c' is the *operation* code. ' 06' is the *sub-command index* (*ii*) for ***Select Protocol***. ' *st*' is the *stream* id digit (0=the ONLY acceptable entry). ' *pro*' is the *protocol* id digit (1=UDP/IP, 0=TCP/IP) ' *remport*' is an optional *remote port number* to which each UDP stream is directed in the host (port 9000 is the default if unspecified). It is ignored if *pro*=0. ' *ipaddr*' is an optional host *IP address* to which each UDP stream is directed (default is the host IP address per current TCP connection that sent this command). It is ignored if *pro*=0. NOTE: all parameters separated by a *space*.
Response	"A" or "Nxx" 'A' for *acknowledge*; 'N' for *negative acknowledge, error code xx*.

This command sets the protocol by which every configured autonomous stream is delivered to the host. **It must be executed after streams are configured, but before they are enabled.**

By default, streams are delivered via the same TCP/IP protocol used to receive commands from host (i.e., via the existing TCP/IP connection used to send this command). However, for special circumstances, all autonomous streams may be delivered to the host via the (less reliable but lower-overhead) UDP/IP protocol instead. This command is required only when UDP/IP is to be used. It also can restore the default protocol (to TCP/IP) once it has been changed. The TCP/IP version of the command ignores the optional (*pro* and *ipaddr*) parameters, which have meaning only to the UDP/IP protocol.

Though the command has a *stream* parameter, it is currently limited to changing the protocol of *all defined streams* at the same time (i.e., parameter *st* **must** be = 0, meaning *all configured* streams).

The optional *remport* parameter may be any value in the range 1024 to 65535. However, *remport* = 7001 should be avoided, since NetScanner modules emit responses to that port, and most host programs (like NUSS) should have a UDP socket already bound to that port for receiving these special responses. The choice of *remport* parameter will affect the way host software must handle multiple modules sending streams. If every module uses the same port, then a single host socket can be bound to that port to receive all responses from all modules. The remote IP address, given to that socket, at time of receipt of the stream's datagram, will identify the particular module who sent the datagram. Alternately, every module may be given a unique *remport* number, requiring that a host program bind a unique UDP socket to each of these unique *remport* numbers. Then, when a particular socket receives a UDP datagram to its unique port, the module sending it is automatically identified.

The optional *ipaddr* parameter is normally unspecified, causing it to default to use the IP address of the current TCP/IP connection. That way the host need not have to be aware of its own IP address. This parameter is provided in case a special host has multiple network interfaces and wants to use more than one. When used, *ipaddr* requires four dotted numeric fields (*d.d.d.d*). Each *d* is a 1-3 digit decimal number in the range 0-255. The *ipaddr* = 255.255.255.255 is best avoided, unless the UDP datagrams of streams are to be broadcast to all network nodes.

Example:

Configure all streams to be delivered via UDP/IP protocol. Host expects the UDP datagrams to arrive via port 7500. The IP Address of the current TCP/IP connection is also used to send each UDP datagram.

"**c 06 0 1 7500**"

Read response:

"**A**"

Configure all streams to be delivered via the default TCP/IP protocol.

"**c 06 0 0**"

Read response:

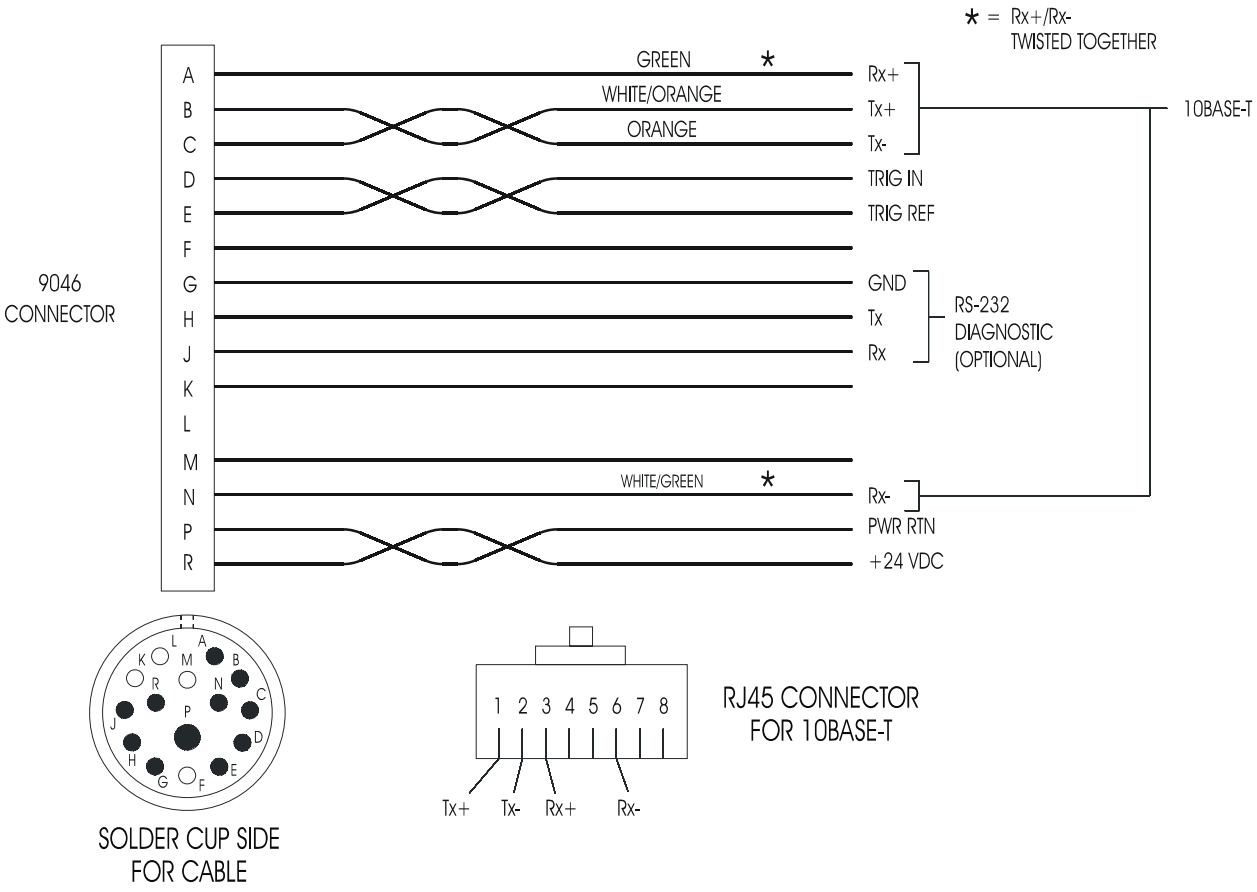
"**A**"

NUSS does not normally use this command since it generates and processes TCP/IP streams exclusively with a unique Socket allocated for each "connected" module. Also, you may NOT execute it manually via the 'Execute Script' or 'Execute Commands' functions of NUSS - as those programs have no facilities to automatically allocate one or more UDP Sockets to receive the data. NUSS expects streams to arrive only via its pre-allocated TCP/IP Socket ports allocated when a module is 'Connected' by NUSS. Only NUSS responses are expected and can be handled by a single UDP Socket NUSS allocates at startup.

This page intentionally left blank.

Appendix A

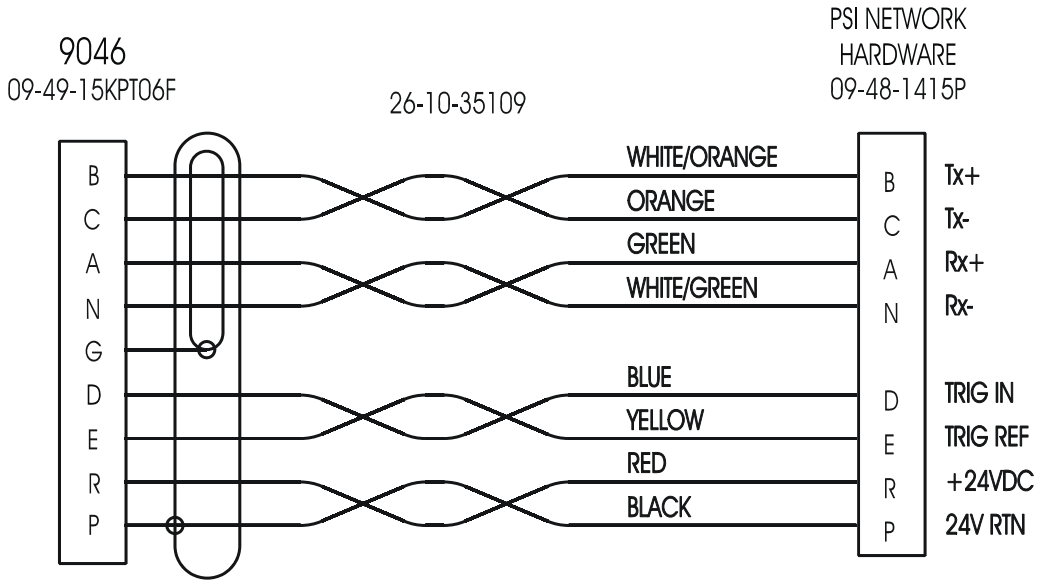
Cable Diagrams



NOTES:

1. WIRE COLORS REFLECT STANDARD RJ45 WIRE ASSIGNMENTS.
2. 10BASE-T SIGNAL WIRES SHOULD MEET CATAGORY 3 OR HIGHER SPECIFICATIONS.
3. POWER AND POWER RETURN CONDUCTORS ARE RECOMMENDED TO BE 18 OR 20 AWG.

Model 9046 Ethernet Interface Cable
P/N 9080



Model 9046 Ethernet Interface Cable
P/N 9080

Pressure Systems, Inc.
34 Research Drive
Hampton, VA 23666
USA
Phone: (757) 865-1243
Toll Free: (800) 328-3665
Fax: (757) 865-8744
E-mail: sales@PressureSystems.com